# Harnessing Wireless Traffic is an Effective Way to Improve Mobile Internet Performance

Richard S.L. Wu, Wilfred W.K. Lin and Allan K.Y. Wong
Department of Computing, Hong Kong Polytechnic University
Hunghom, Kowloon, Hong Kong SAR, P.R. China
Emails: csslwu@comp.polyu.edu.hk, cswklin@comp.polyu.edu.hk, csalwong@comp.polyu.edu.hk

**Abstract**

Today mobile Internet, which is the "wireline and wireless (W&W)" environment, is a reality. Although broadband wireless technologies may replace the cable communication ones in the future, we have to work with the W&W framework for quite a while. The introduction of wireless communication in the Internet environment, broadband or not, has brought a new problem. That is, the traffic pattern can cause instability to the system. To ward off the traffic ill effects a novel real-time traffic pattern detection (RTPD) technique is proposed. If it is incorporated into a logical entity, the latter can use the detected result to reconfigure on the fly and becomes immune to traffic ill effects.

Keywords: mobile Internet, wireless communication, real-time traffic pattern detection, reconfiguration

## 1. Introduction

By Moore's Law the VLSI (Very Large Scale Integration) hardware speed in MIPS has already hit the limit of physics and the natural solution is to go distributed. As a result computation speedup is gained from the distributed hardware parallelism. This makes the Internet an attractive platform for distributed processing and this is "megacomputing" in Ted Lewis's term [1]. Undoubtedly distributed computing is the reality, and it would become our future provided that network speed grows faster than the VLSI speed. Indeed, VLSI and network speeds crossed over around the year 2004 [1] and megacomputing is no longer far fetched. In megacomputing the network transmits much faster than the CPU can handle. Using the M/M/1 model as an example, the speedup S gained by distributed parallelism or megacomputing can be expressed as

$$S = \frac{W_1}{W_n} = \frac{(1-\rho/n)}{(1-\rho)} \text{ [1],}$$

where $W_1$ the response time of the M/M/1 model, $W_n$ the response time of the M/M/n model with $n$ as the number of parallel Internet nodes, and $\rho$ the system utilization. In early 1990s the construction of a megacomputer was considered conceptually possible because of the quantum leap of network speed, exemplified by the pure optic fibers. These fibers work with ideal light pulses known as *solitons* [2]. The speed of wireless technologies had improved since, and the IEEE Working Group 802.16 standards for Broadband Wireless Access (BWA) tried to tap the 10 to 66 gigahertz (GHz) range that includes licensed and unlicensed spectrums. Meanwhile different products keep appearing and claim to operate in that range. Yet, they have interoperability problems especially in the medium access control (MAC) and physical (PHY) levels [3]. To resolve these problems IEEE 802.16 and the associated industry consortium, WiMax created profiles; adherence to a profile means interoperability among vendors. The MiMax focuses on frequencies below 11 GHz. In fact, the 802.16 effort includes finding a quickly deployable alternative to replace cabled access network technologies such as fiber optic links and digital subscriber lines (DSL). The advantage of the replacement is that robust wireless access can reach a broad geographic area without the need of a costly infrastructure (e.g. the need to dig and lay cable links). Despite its potential, broadband wireless communication presently suffers from the following problems: a) up to 25 percent of a cell would have communication blackout for

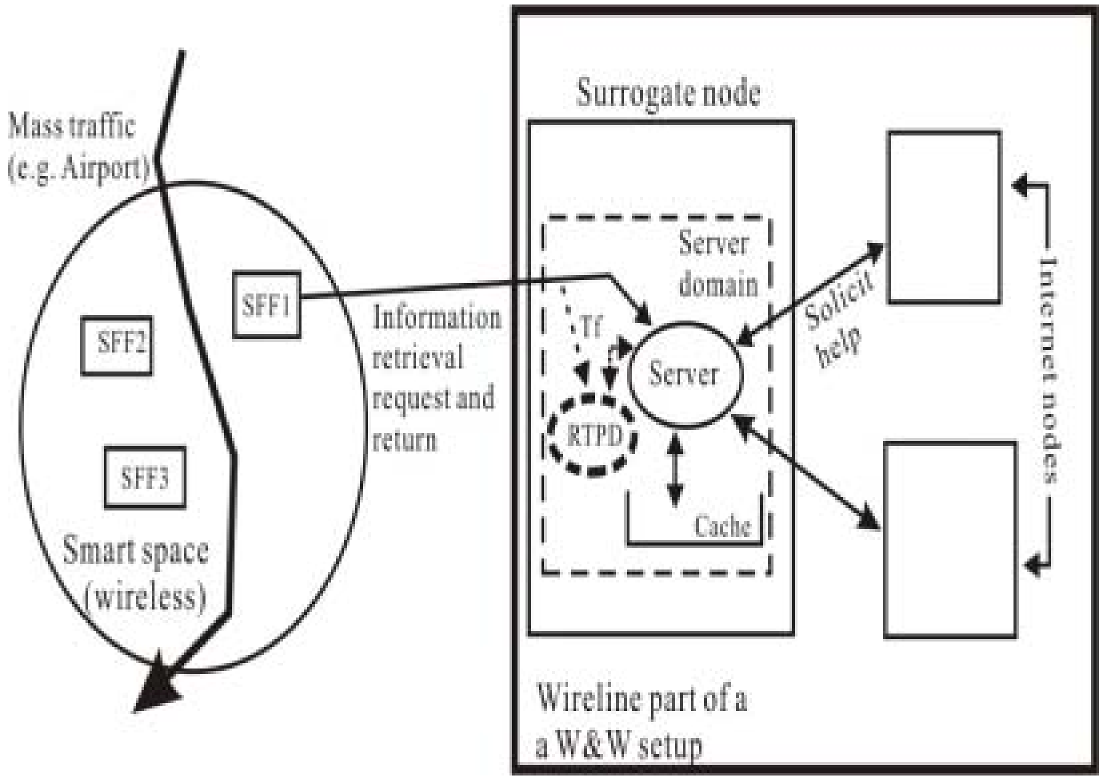


**Figure 1. A dynamic cache size tuning system in a W&W environment**

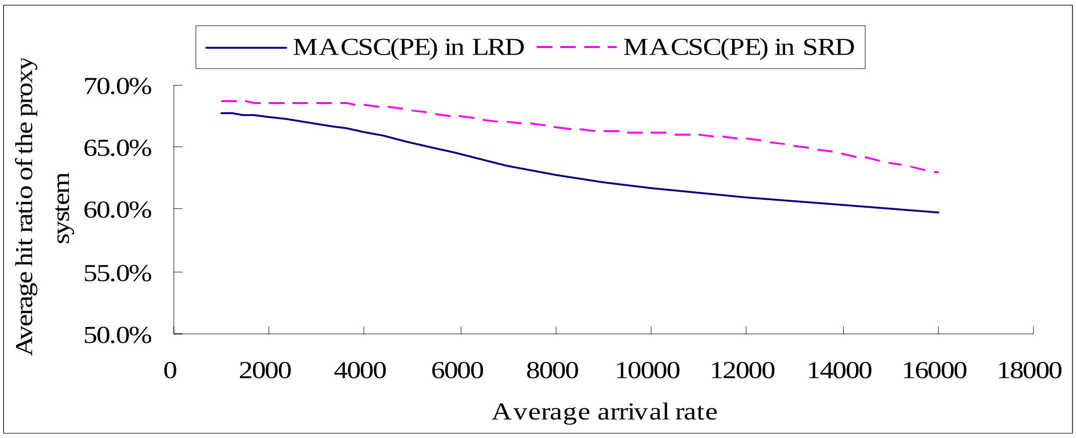


**Figure 2. DCSTS (i.e. MACSC(PE)) hit ratio versus IAT for the wireless LRD and SRD traces**

single-antenna users, b) poor reception if the user is traveling more than 80 miles an hour, and c) high loss rate. In fact, the high loss rate problem can be resolved by adapting the automatic repeat request (ARQ) scheme that recovers losses by retransmissions [3,4]. Recent findings, however, show that reliability of the channel between two interacting users should include the user level as well. Besides, traffic patterns in the channel can cause system instability [5]. This paper proposes *real-time traffic pattern detection* (RTPD) as a means to eliminate traffic ill effects over the mobile Internet, which is a “wireline + wireless (W&W)” environment for client/server interactions. The accuracy of the RPTD mechanism is independent of traffic pattern and any wireless technology. In fact, the W&W operation is the basis for pervasive computing [6]. The users in this environment are mobile and travel from one wireless cell (called a *smart space*) to another. While traveling they interact with the environment through their SFF (small form factor) devices (e.g. PDA).

## 2. Problem Definition and Related Work

The ill effects of traffic patterns on system stability were reported in the dynamic cache size tuning system (DCSTS) [7]. The DCSTS case is shown in Figure 1 for the purpose of explanation, and we will use this case to demonstrate how the proposed RTPD mechanism helps rectify the problem. The original DCSTS with respect to Figure 1 lacks the RTPD (encircled by a dotted line), which samples the request traffic indicated by *Tf*. The RTPD, if incorporated, would strive to maintain the given cache hit ratio to shorten the information retrieval time for the user (e.g. SFF1). This is achieved because when the data can be found in the local cache, then cyber foraging, which enlists help from other Internet nodes in the wireline part of W&W, can be obviated. For example, if the cache hit ratio were to be maintained at 70 % or 0.7 and the roundtrip time for cyber foraging were 20 times of that between SFF1 and the server in the surrogate node, the speedup due to caching $S_{cache}$ would be

$$S_{cache} = 21 / [1 + (0.7 * 0 + 0.3 * 20)] = 3$$

The performance data of the original DCSTS (without RTPD) confirms that the cache hit ratio fluctuates due to traffic ill effects. Figure 2 is the plot of the average cache hit ratio of the original DCSTS (called MACSC(PE) in [7]) versus the inter-arrival times (IAT) for LRD (long-range dependence) traffic patterns such as heavy-tailed and self-similar, as well as SRD (short-range dependence) cases with Markovian patterns.

With respect to Figure 1 the traffic pattern for the information retrieval requests initiated from the smart space ties in with the mass traffic pattern passing through it at the time. The reason is that the users would switch on their SFF devices to make use of the smart space facilities. The ill effects of the mass traffic indirectly produce perturbations in the cache hit ratio of the server in the surrogate node (e.g. Figure 2). It was confirmed by previous work [7] that the negative impact on performance by LRD is generally more severe than SRD. The logical solution to eliminate traffic ill effects is to let a system (e.g. DCSTS) have the ability to detect/identify traffic pattern on the fly. With the detected result the system could reconfigure itself in a dynamic manner to eliminate the traffic ill effects. The proposed RTPD technique helps achieve such a goal.

### 2.1 The MACSC(PE) Essence

The MACSC (Model for Adaptive Cache Size Control) framework leverages the ratios of two successive standard deviations of the relative popularity profile (RPP) of a set of data objects. A data object is relatively more popular because it has higher access frequencies. If the RPP is a plot with the more popular objects placed in the central region, a bell-shaped curve/distribution is formed as shown by Figure 3. Actually Figure 3 shows the shape of the RPP in different time points (i.e. A, B, and C). The change in the RPP shape is caused by the shift of user preference toward different data objects at different times. If the cache hit ratio is one standard deviation (i.e. 68.4%) for curve A, then the same cache size does not yield the same hit ratio for curves B and C because they need a large cache size to accommodate “their one standard deviation of data objects”. In the MACSC framework the cache size is adjusted by equation (2.1), where $SD_{thiscycle}$ the standard deviation of the current RPP, $SD_{lastcycle}$ as that computed for the last RPP, $CS_{adjusted}$ the adjusted cache size and $CS_{present}$ the present cache size.

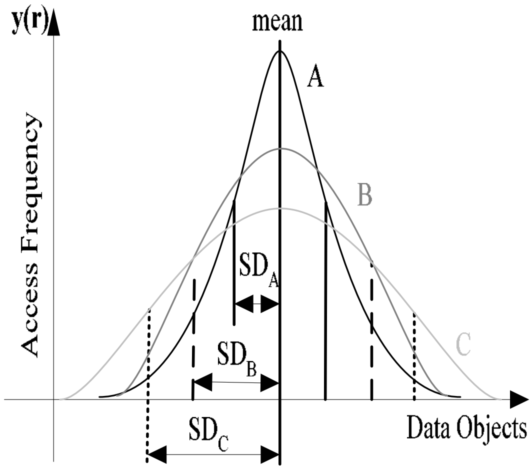


**Figure 3. The relative popularity profiles for the same set of data objects at three time points**

$$CS_{adjsuted} = \frac{SD_{thiscycle}}{SD_{lastcycle}} * CS_{present} \qquad (2.1)$$

MACSC(PE) is one of the implementations of the MACSC framework. This implementation uses the point-estimate (PE) approach, which is a statistical method to compute $SD_{thiscycle}$ [7]. This method is derived from the Central Limit Theorem and its accuracy is therefore waveform independent.

## 3. The Real-time Traffic Pattern Detection Solution (RTPD)

The novel real-time traffic pattern detection or RTPD solution proposed in this paper is based on the conventional R/S (rescaled adjusted statistics) approach, which is defined as

$$\frac{R}{S} = \frac{\max(W_i : i = 1,2,..,n) - \min(W_i : i - 1,2,..,n)}{\sqrt{\mathrm{var}(X)}}$$

where $W_i = \sum_{k=1}^{i} (X_k)$ for i=1,2…$n$

and $\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$ [8].

The conventional R/S form, however, is unsuitable for real-time application because of the unpredictable $k$ value. When $k$ is large the time needed for sampling the required data items on-line can be excessively long, depending on the average IAT in the period. A large $k$ value could lead to deleterious effects in practice because by the time the remedial solution is ready, the problem has already gone. Using the computed solution to remedy a ghost problem ends up with undesirable/deleterious results. The useful feature of R/S, $R/S \approx (n/2)^H$ depends on a statistically reasonable $n$ value, which is tied to $k$. The $H$ (Hurst) value is indicative: $0 < H < 0.5$ for SRD and $0.5 < H < 1$ for LRD.

The important step to convert the conventional R/S approach for real-time application is to make $k$ predictable. In the RTPD framework this is achieved by using the real-time $M^3RT$ IEPM (Internet End-to-End Performance Measurement) technique [9], which estimates the mean of any waveform quickly. The $M^3RT$ accuracy is independent of the waveform type because it is derived from the Central Limit Theorem. The $M^3RT$ essence is represented by $M_j = \frac{P * M_{j-1} + \sum_{l}^{f-1} m_l^j}{P + f}$, where P is the damping factor, $M_{j-1}$ the mean estimated in the last cycle as feedback to the current $j^{th}$ cycle, $m^j$ the $l^{th}$ sample in the $j^{th}$ $M_j$ estimation cycle among the total of $f$ data items to be sampled. The range for fast convergence is $9 < f < 16$ and $f = 14$ is the fastest and is therefore adopted in RTPD.

In the real-time R/S approach (or R-R/S) the value $\overline{X}$ is replaced by $M_j$. The R-R/S computation is the core of the RTPD framework. If it is incorporated into the construct of a logical server, then the latter becomes reconfigurable in the sense that it uses the detected traffic pattern to reconfigure itself to ward off the traffic ill effects. In this paper the experimental results of the "RTPD + MACSC(PE)" combination or simply called RTPD/MACSC(PE) in a wireless environment will be presented. The wireless traffic in the experiments includes broadband sources [10]. Repeated timing analysis reveals that the R-R/S needs an average of only 890 intrinsic clock cycles to execute and is therefore immensely suitable for real-time applications. It is intrinsic because the data items for computing R/S are immediately usable from the pre-collected traces. In the real-time environment the R/S computation time is much longer because the IAT delay in between two samples. The "$f$-1" number of IAT delays is inevitable for R-R/S but it is much better than the conventional R/S with an unpredictable $k$, i.e., $k$ IAT delays. The R-R/S execution time down-scales proportionately with the clock rate of the platform because the physical time PT is $PT = 890/PLF\_Hz$ , where $PLF\_Hz$ is the platform speed in hertz.

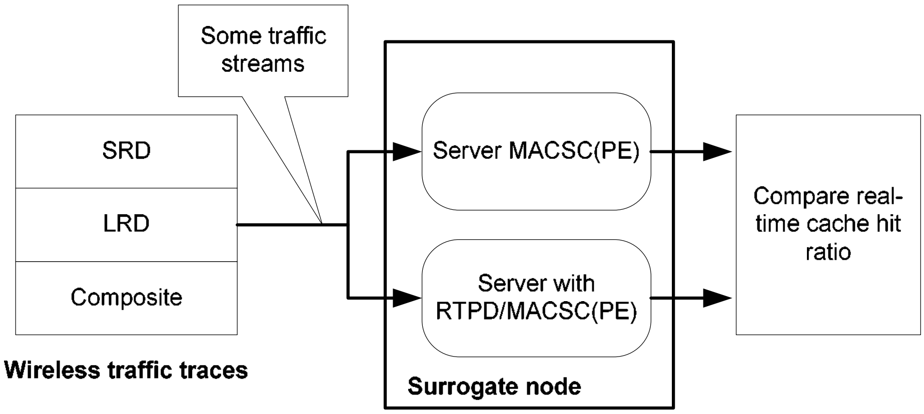


**Figure 4. The verification environment**

## 4. Verification Results

The verification experiment made use of the MACSC(PE) framework. The RTPD solution is incorporated to make this framework into a reconfigurable DCSTS, namely, RTPD/MACSC(PE). The experimental setup is shown in Figure 4. The MACSC(PE) and the RTPD/MACSC(PE) modules are bombarded by the same wireless traffic [10]. The results from different experiments show that the RTPD/MACSC(PE) always maintains at least the given cache hit ratio (i.e. one standard deviation (i.e. 68.4%) for the experiments) in a consistent manner under all traffic conditions. The cache hit ratio, which is supposedly to be maintained by the unsupported MACSC(PE) deteriorates with increased IAT. Figure 5 is a test case in which the wireless traffic trace is composite in the sense that SRD and LRD pattern sections interleave repeatedly.

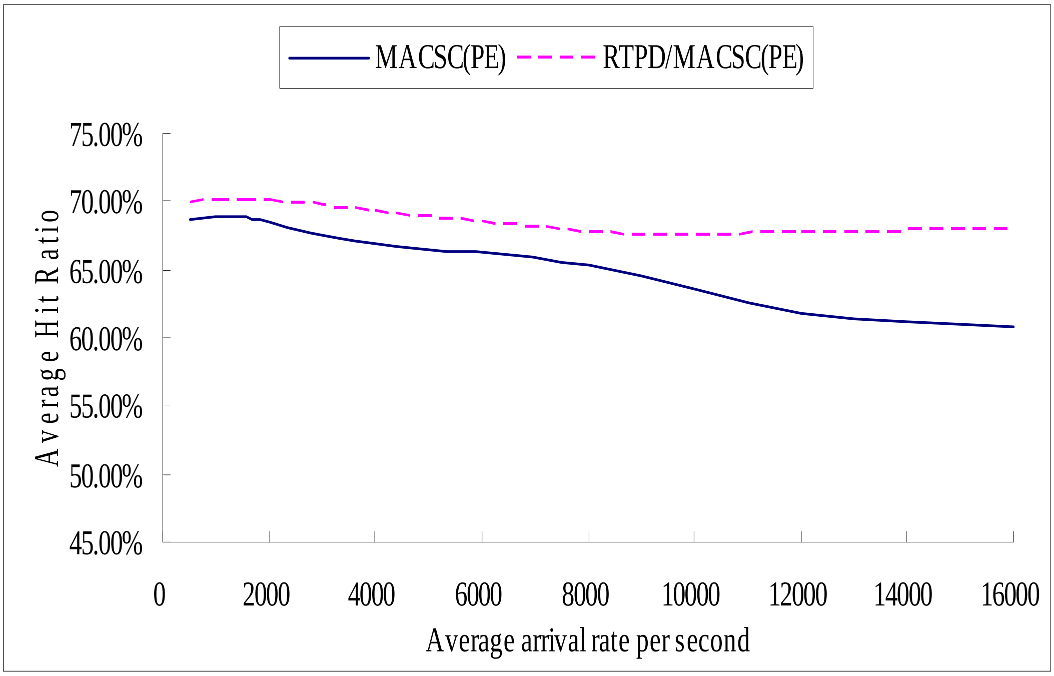


**Figure 5. RTPD/MACSC(PE) yields higher hit ratio than MSCSC(PE) for a composite wireless trace**

## 5. Conclusion

Mobile Internet, which is basically a “wireline + wireless (W&W)” environment, is the reality today. Although the potential of replacing cable based communication technologies by broadband wireless ones such as those being standardized by the IEEE 802.16 standards is there, complete replacement still has a long way to go. For example, international research energy (e.g. WiMax) is still concentrated only on bandwidths lower than 11 GHz. In fact, the availability of untapped broadband capability is up to 66 GHz. In the W&W environment the wireless convenience has brought new problems. For example, traffic ill effects can cause system instability. In this paper a novel RTPD technique is proposed for detecting SRD and LRD traffic patterns on the fly. With this capability mobile Internet applications such as the MACSC(PE) can reconfigure itself to ward off traffic ill effects. The verification results with the novel RTPD/MACSC(PE) framework indicate that RTPD is indeed an effective solution in this aspect. In the near future the RTPD solution will be applied to different applications on the mobile Internet so that its potency for generic applications can be validated.

## 6. Acknowledgement

The authors thank the Department of Computing and the Hong Kong Polytechnic University for the A-PG51 research grant.

## 7. References

[1] T. Lewis, The Next $10{,}000_2$ Years: Part 1, IEEE Computer, 29(4), April 1996, 64 - 70

[2] R.W. Keyes, The Future of Transistor, Scientific America, June 1993

[3] A. Ghosh, D.R. Wolter, J.G. Andrews and R. Chen, Broadband Wireless Access with WiMax/802.16: Current Performance Benchmarks and Future Potential, IEEE Communications Magazine, February, 2005, 129 - 136

[4] S.Lin and P. Yu, A Hybrid ARQ Scheme with Parity Retransmission for Error Control of Satellite Channels, IEEE Transactions of Communications, 30(7) July 1982, 1701 - 1719

[5] Wilfred W. K. Lin, Allan K. Y. Wong, Richard S.L. Wu, and Tharam S. Dillon, A Novel Real-Time Self-Similar Traffic Detector/Filter to Improve the Reliability of a TCP Based End-to-End Client/Server Interaction Path for Shorter Roundtrip Time, Proc. of the International Conference on E-business and Telecommunication Networks (ICETE’05), Reading UK, vol. 1, October 2005, 94-101

[6] D. Garlan, D.P. Siewiorek, A. Smailagic and P. Steenkiste, Project Aura: Toward Distraction-free Pervasive Computing, IEEE Pervasive Computing, 1(2), April 2002, 22 - 31

[7] Richard S.L. Wu, Allan K.Y. Wong and Tharam S. Dillon, E-MACSC: A Novel Dynamic Cache Tuning Technique to Reduce Information Retrieval Roundtrip Time over the Internet, Computer Communications, 2005 (in press)

[8] S. Molnár, T.D. Dang and A. Vidacs, Heavy-Tailedness, Long-Range Dependence and Self-Similarity in Data Traffic, Proceedings of 7th International Conference on Telecommunication Systems, Modelling and Analysis, March 1999, Nashville, USA, 18 – 21

[9] Allan K.Y. Wong, Tharam S. Dillon, Wilfred W.K. Lin and T.W. Ip, M2RT: A Tool Developed for Predicting the Mean Message Response Time for Internet Channels, Computer Networks (and ISDN Systems), vol. 36, 2001, 557-577

[10] MosquitoNet: The Mobile Computing Group at Standford University, http://mosquitonet.stanford.edu